What is Git??

- **Git** is an **open-source distributed version control system**.
- It is designed to handle minor to major projects with high speed and efficiency. It is developed to co-ordinate the work among the developers.
- The version control allows us to track and work together with our team members at the same workspace.

Features of Git

- Some remarkable features of Git are as follows:

Open Source

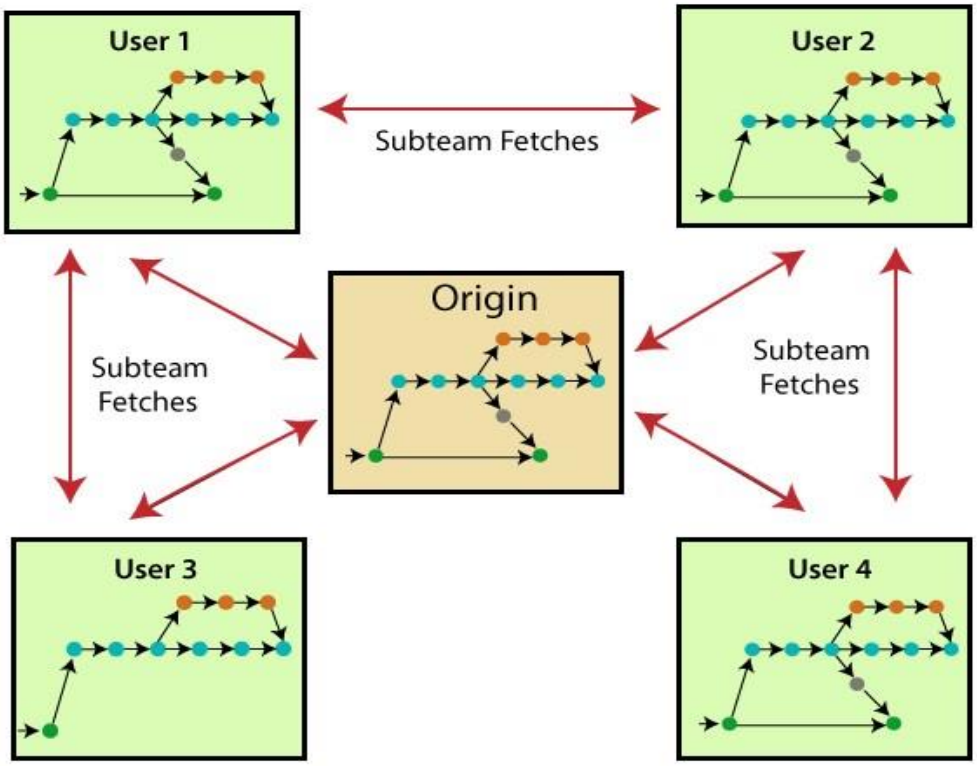
- Git is an open-source tool. It is released under the GPL (General Public License) license.

Scalable

- Git is scalable, which means when the number of users increases, the Git can easily handle such situations.

Distributed

- One of Git's great features is that it is distributed. Distributed means that instead of switching the project to another machine, we can create a "clone" of the entire repository.



Security

- Git is secure. It uses the SHA1 (Secure Hash Function) to name and identify objects within its repository.
- Files and commits are checked and retrieved by its checksum at the time of checkout.
- It stores its history in such a way that the ID of commits depends upon the complete development history leading up to that commit. • Once it is published, one cannot make changes to its old version.

Speed

- Git is very fast, so it can complete all the tasks in a while.
- Most of the git operations are done on the local repository, so it provides a huge speed.
- Also, a centralized version control system continually communicates with a server somewhere.

Supports non-linear development

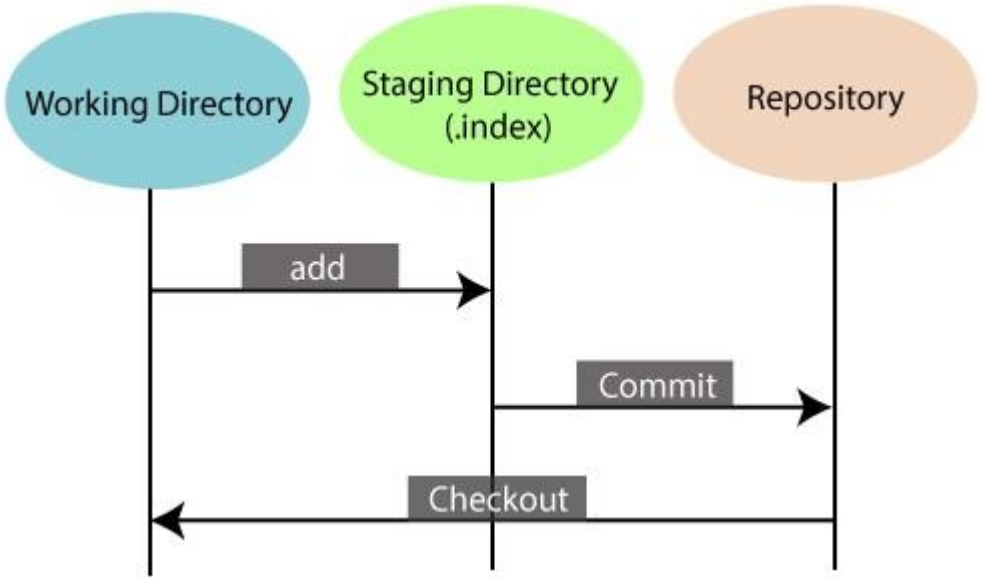
- Branching and merging are the great features of Git, which makes it different from the other SCM tools.
- Git allows the creation of multiple branches without affecting each other.
- Perform tasks like creation, deletion, and merging on branches, and these tasks take a few seconds only.

Data Assurance

- The Git data model ensures the cryptographic integrity of every unit of our project. It provides a unique commit ID to every commit through a SHA algorithm.
- We can retrieve and update the commit-by-commit ID. Most of the centralized version control systems do not provide such integrity by default.

Staging Area

- The Staging area is also a unique functionality of Git.
- It can be considered as a preview of our next commit, moreover, an intermediate area where commits can be formatted and reviewed before completion.



What is GitHub?

- GitHub is a Git repository hosting service.
- GitHub also facilitates with many of its features, such as access control and collaboration. It provides a Web-based graphical interface.
- GitHub is an American company.
- It hosts source code of your project in the form of different programming languages and keeps track of the various changes made by programmers.
- It offers both **distributed version control and source code management (SCM)** functionality of Git.
- It also facilitates with some collaboration features such as bug tracking, feature requests, task management for every project.

Some of its significant features are as follows.

- Collaboration
- Integrated issue and bug tracking
- Graphical representation of branches
- Git repositories hosting
- Project management
- Team management

- Code hosting
- Track and assign tasks
- Conversations Git

Environment Setup

The Git config command

- Git supports a command called git config that lets you get and set configuration variables that control all facets of how Git looks and operates.
- It is used to set Git configuration values on a global or local project level.
- Setting user.name and user.email are the necessary configuration options as your name and email will show up in your commit messages.

Example:

$ git config --global user.name "Himanshu Dubey"

$ git config --global user.email "himanshudubey481@gmail.com"

$ git config --global core.editor Vim

$ git config -list

Git Tools

Basic Git Commands

Here is a list of most essential Git commands that are used daily.

- Git Config command
- Git init command
- Git clone command
- Git add command
- Git commit command
- Git status command
- Git push Command
- Git pull command
- Git Branch Command
- Git Merge Command
- Git log command
- Git remote command

-

Git Commands

- git config --global user.name "Venkataram"
- git config --global user.email “venkatcmy1986@gmail.com”
- ssh-keygen -t ed25519 -C "venkatcmy1986@example.com"
- clip < ~/.ssh/id_ed25519.pub
- git clone git@github.com:venkatcmy1986/DXC-Test.git git .
- gitignore LICENSE README.md
- Add some files locally or in github
- GitStatus command **`git status`**
- `Git Restore command.`
- `Git Add command git add filename.`
- `Git Commit command.` **`git commit -m “enter comments”.`**
- Git push command
- Git log command